

Introduction to HPC Using zcluster at GACRC

On-class PBIO/BINF8350

Georgia Advanced Computing Resource Center

University of Georgia

Zhuofei Hou, HPC Trainer

zhuofei@uga.edu

Outline

- What is GACRC?
- What is HPC Concept?
- What is zcluster?
- How does zcluster operate?
- How to work with zcluster?

What is GACRC?

Who Are We?

- Georgia **A**dvanced **C**omputing **R**esource **C**enter
- Collaboration between the Office of Vice President for Research (**OVPR**) and the Office of the Vice President for Information Technology (**OVPIIT**)
- Guided by a faculty advisory committee (GACRC-AC)

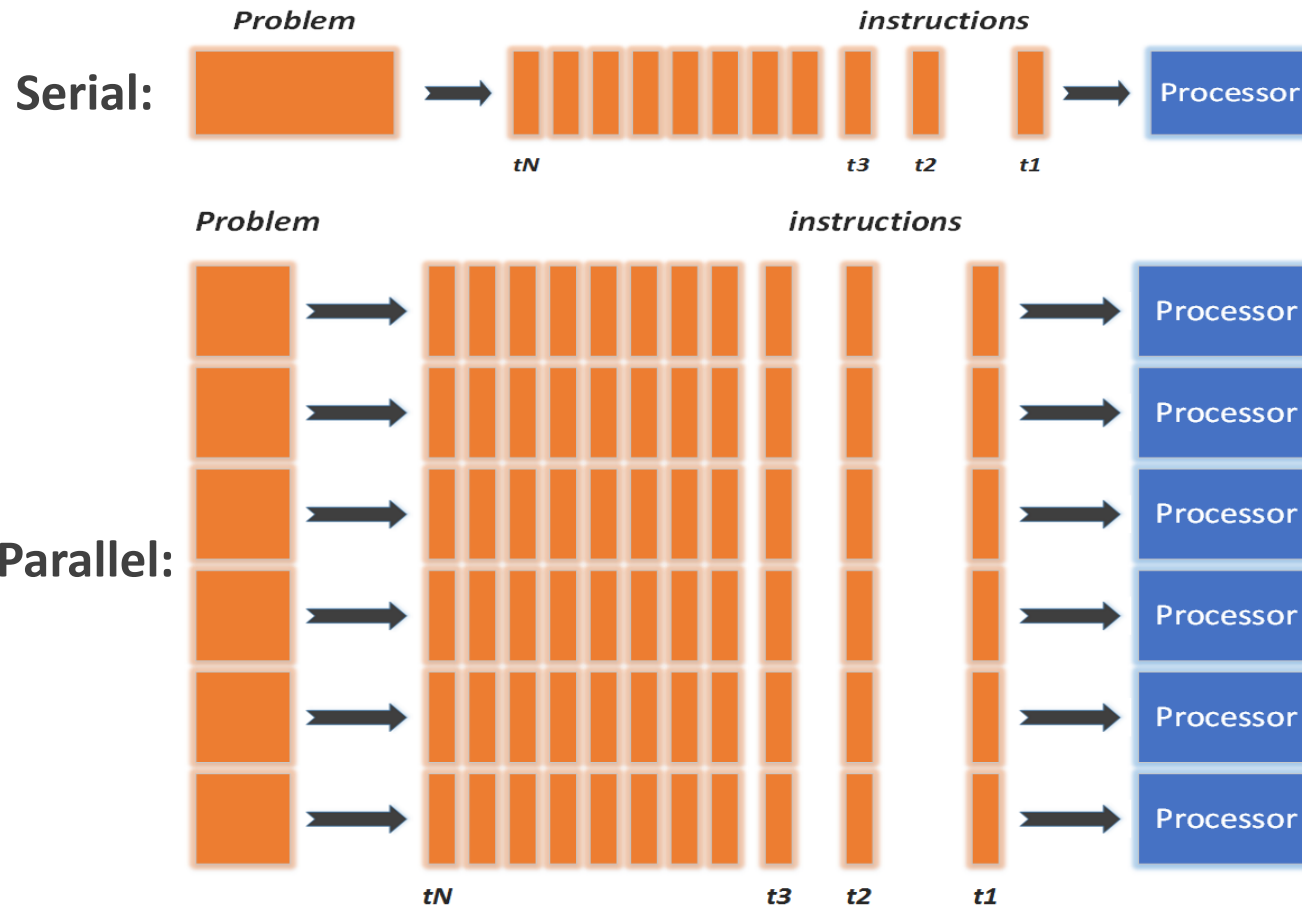
Why Are We Here?

- To provide computing hardware and network infrastructure in support of high-performance computing (**HPC**) at UGA

Where Are We?

- <http://gacrc.uga.edu> (Web) <http://wiki.gacrc.uga.edu> (Wiki)
- <http://gacrc.uga.edu/help/> (Web Help)
- https://wiki.gacrc.uga.edu/wiki/Getting_Help (Wiki Help)

Concept of High Performance Computing (HPC)



- ✓ **Serial** problem can not be broken
- ✓ *Discrete* instructions executed *sequentially*
- ✓ Only *1* instruction executed at any moment on a *single* processor

- ✓ Problem broken into *parallel* parts can be solved *concurrently*
- ✓ Instructions executed *simultaneously* on *multiply* processors
- ✓ Synchronization/communication employed
- ✓ **Shared-memory multithreaded job** or **MPI job** (Message Passing Interface)

What is zcluster?

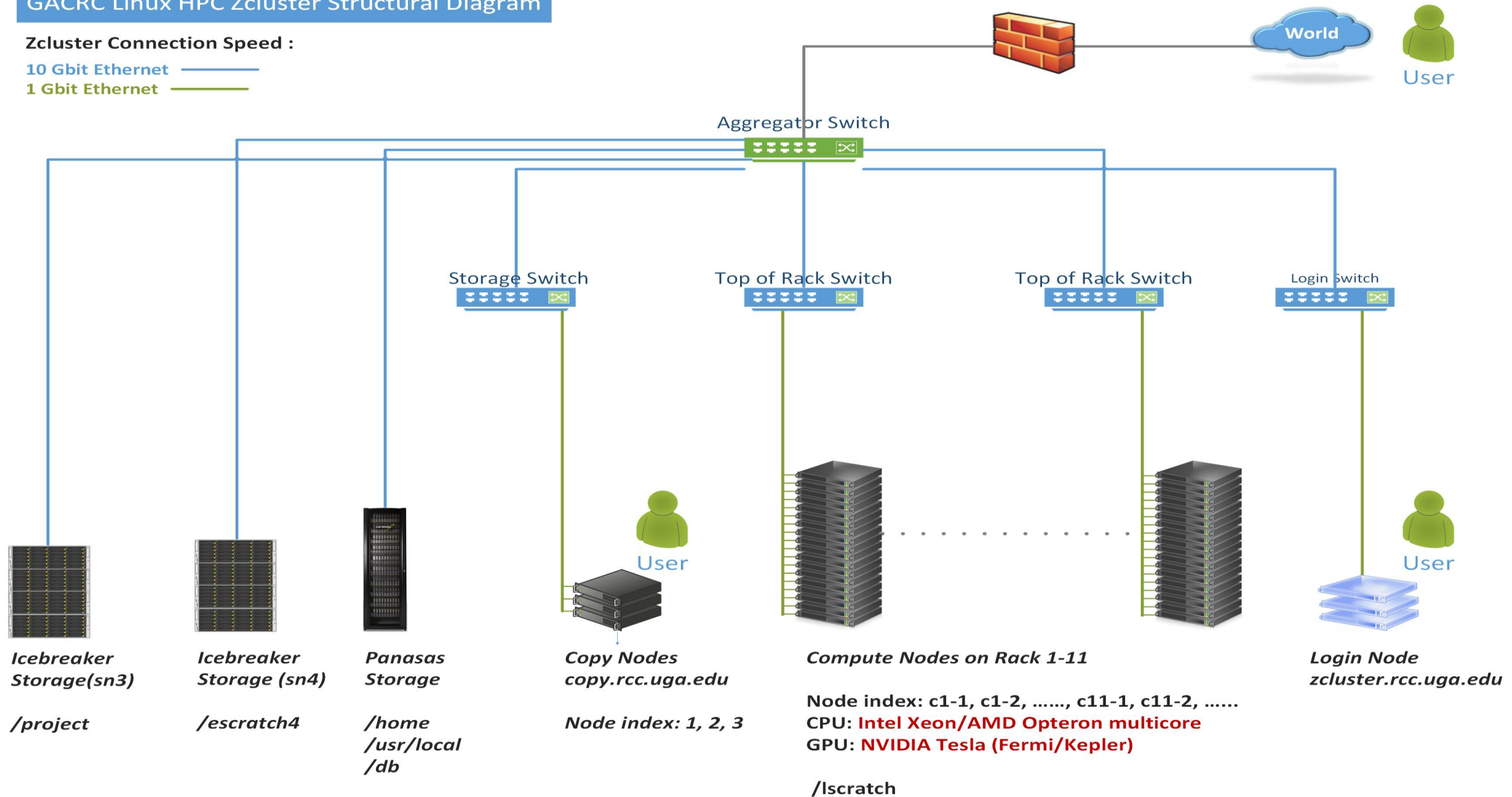
- Cluster Structural Diagram
- General Information
- Computing Resources
- Storage Environment

GACRC Linux HPC Zcluster Structural Diagram

Zcluster Connection Speed :


10 Gbit Ethernet

1 Gbit Ethernet



What is zcluster – General Information

GACRC zcluster is a Linux high performance computing (HPC) cluster:

- Operating System: 64-bit Red Hat Enterprise Linux 5 (RHEL 5)
- Login Node: zcluster.rcc.uga.edu ^{qlogin}  Interactive Node: compute-14-7/9
Copy Node: copy.rcc.uga.edu
- Internodal Communication: 1Gbit network
compute nodes ↔ compute nodes
compute nodes ↔ storage systems

What is zcluster – General Information

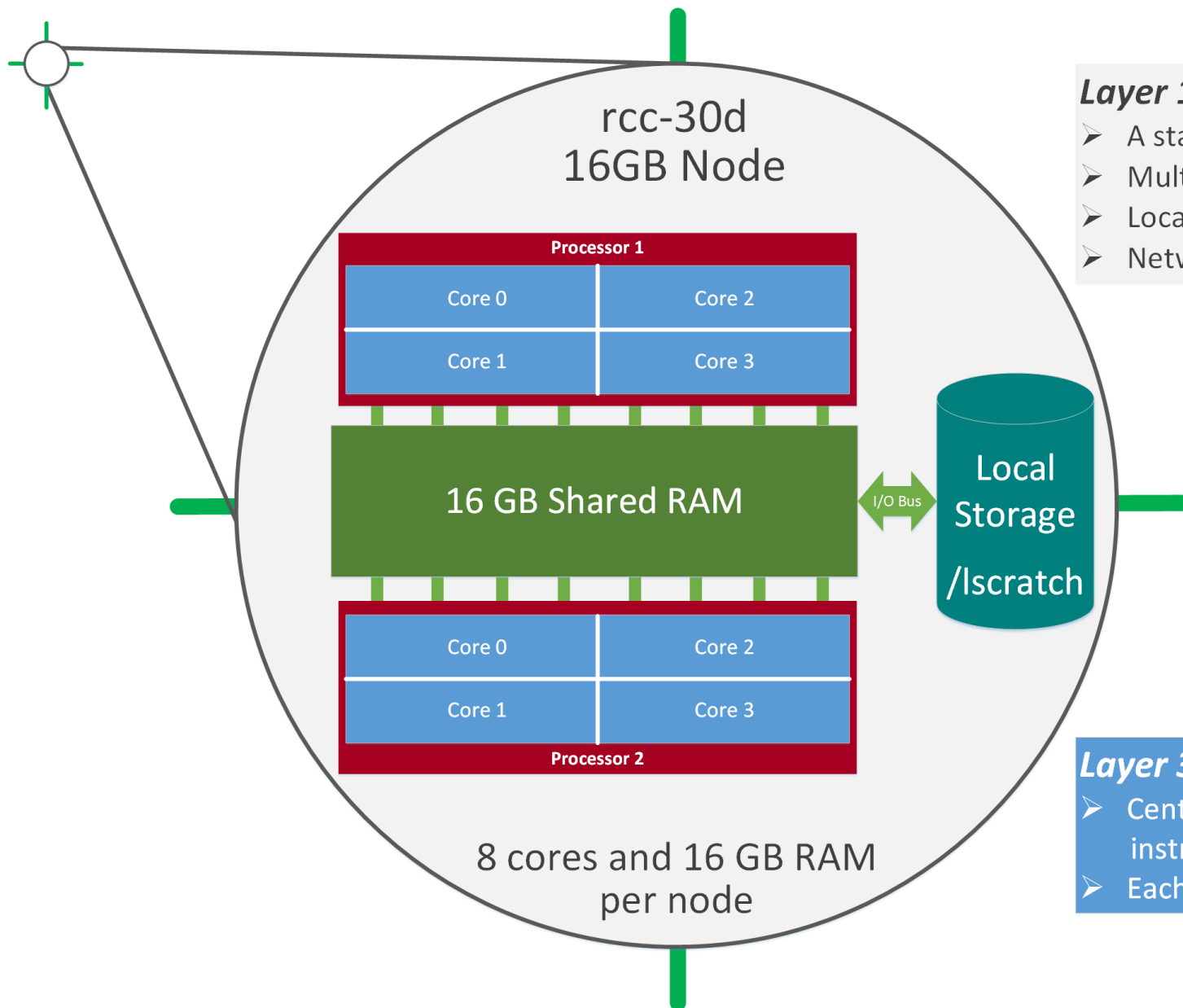
- Batch-queueing System:
 - Jobs can be started (submitted), monitored, and controlled
 - Determine which compute node is the best place to run a job
 - Determine appropriate execution priority for a job to run
- On zcluster: **Sun Grid Engine (SGE)**



What is zcluster – Computing Resources

Queue Type	Queue Name	Nodes	Processor	Cores/Node	RAM(GB)/Node	Cores	NVIDIA GPU
Regular	rcc-30d	45	Intel Xeon	12	48	540	N/A
		150		8	16	1200	
High Memory	rcc-m128-30d	1	Intel Xeon	8	128	8	N/A
		4		8	192	32	
		10		12	256	120	
	rcc-m512-30d	2		32	512	64	
Multi Core	rcc-mc-30d	6	AMD Opteron	32	64	192	N/A
Interactive	interq	2	AMD Opteron	48	132	96	N/A
GPU	rcc-sgpu-30d	2	Intel Xeon	8	48	16	4 Tesla S1070 cards
	rcc-mgpu-30d	2		12	48	24	9 Tesla (Fermi) M2070 cards
	rcc-kgpu-30d	4		12	96	24	32 Tesla (Kepler) K20Xm cards

Total peak performance: 23 Tflops



Layer 1: Node

- A standalone “computer in a box”
- Multiple processors, e.g. 2, sharing memory
- Local disk storage, network interface, etc.
- Networked into a cluster

Layer 2: Processor

- A single computing component
- Multicore processor, e.g. 4 cores

Layer 3: Core

- Central processing unit (CPU) reading and executing instructions independently
- Each core is assigned to a software thread

What is zcluster – Storage Environment

- **Home directory** → `/home/student/pbio8350/s_20`
- Mounted and visible on **all nodes**, with a quota of **~100GB**
 - Any directory on `/home` has **snapshot** backups
 - Taken once a day, and maintained **4 daily** ones and **1 weekly** one
 - Name: **.snapshot**, e.g., `/home/abclab/jsmith/.snapshot`
 - **Completely invisible**, however, user can “cd” into it and then “ls”:

```

zhuofei@zcluster:~$ ls -a
.          .bash_profile  .emacs.d    .fontconfig  .maple_history  MPis      scripts  test.sh
..         .bashrc        .ENV_file   .gnuplot_history  .Mathematica    openMPs   serials  .viminfo
.bash_history  downloads    exe         .history      .mc              .profile  sht      .Xauthority
.bash_logout  .emacs      .flexlmrc   .lessht       .mozilla         Pthreads  .ssh     ← .snapshot is NOT
zhuofei@zcluster:~$ cd .snapshot ← can “cd” into .snapshot
zhuofei@zcluster:~/ .snapshot$ ls ← then “ls” to list its contents
2015.06.21.00.00.01.weekly  2015.06.27.01.00.01.daily  2015.06.28.01.00.01.daily  2015.06.30.01.00.01.daily
2015.06.26.01.00.01.daily  2015.06.28.00.00.01.weekly  2015.06.29.01.00.01.daily
  
```

What is zcluster – Storage Environment

- **Local scratch** → `/lscratch/s_20`
 - On **local disk** of each **compute** node → **node-local storage**
 - rcc-30d 8-core nodes: **~18GB**, rcc-30d 12-core nodes: **~370GB**
 - **No snapshot backup**
 - Usage Suggestion: *If your job writes results to /lscratch, job submission script should move the data to your home or scratch before exit*
- **Ephemeral Scratch** → `/scratch4/s_20/s_20_Jan_15`
 - Create with `make_scratch` command
 - Visible to **all nodes** with a quota of **4TB**
 - **No snapshot backup**
 - To be deleted after **37 days**

What is zcluster – Storage Environment

Filesystem	Role	Quota	Accessible from	Intended Use	Notes
/home/student/pbio8350/s_20	Home	100GB	zcluster.rcc.uga.edu (Login) copy.rcc.uga.edu (Copy) Interactive nodes (Interactive) compute nodes (Compute)	Highly static data being used frequently	Snapshots
/escratch4/s_20	Scratch	4TB		Temporarily storing large data being used by jobs	Auto-deleted in 37 days
/lscratch/s_20	Local Scratch	18 ~ 370GB	Individual compute node	Jobs with heavy disk I/O	User to clean up

- Note:
1. /usr/local : Software installation directory
/db : bioinformatics database installation directory
 2. To login to Interactive nodes, use `qlogin` from Login node

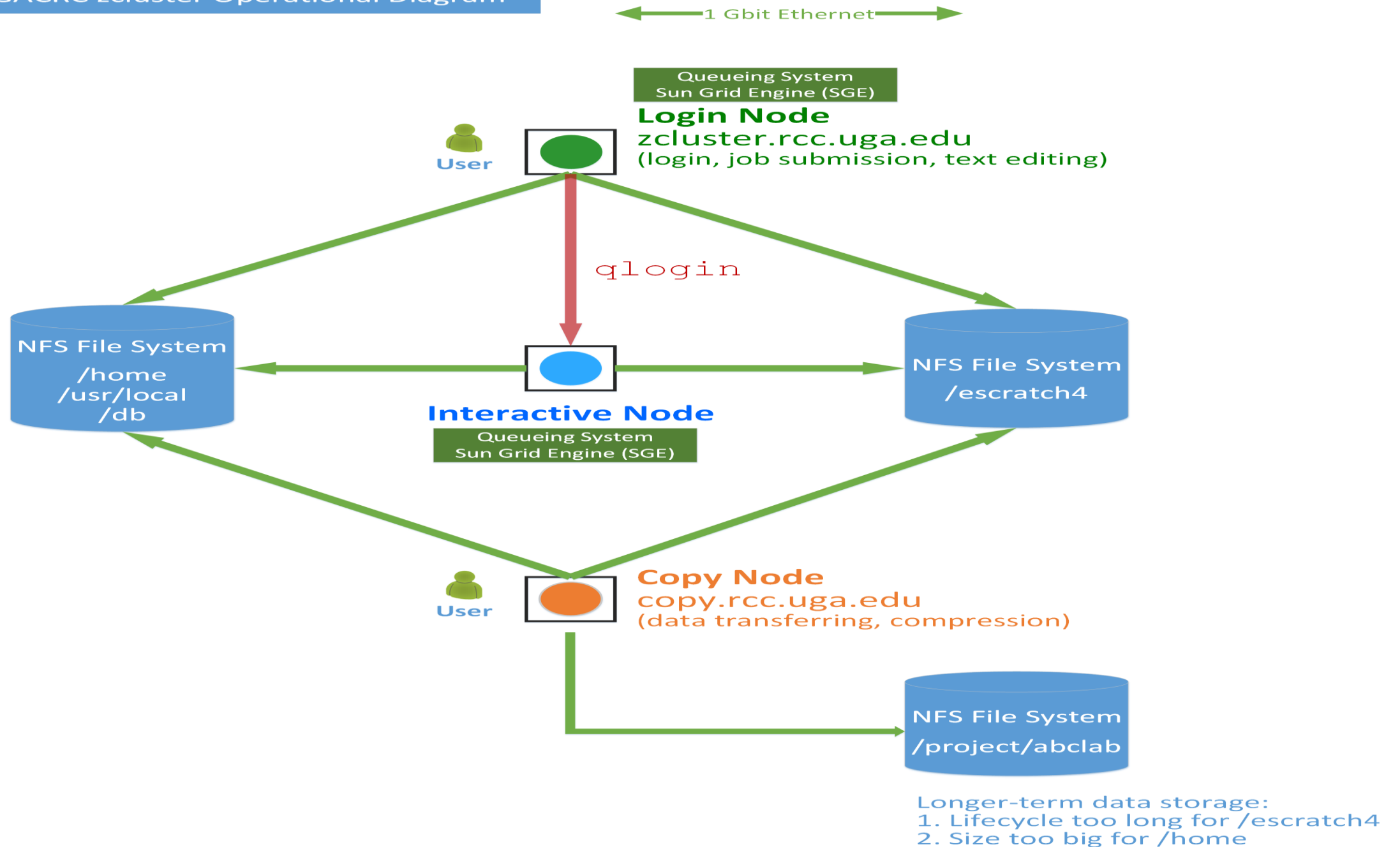
What is zcluster – Storage Environment

	6 Main Function	On/From-Node	Related Filesystem
➔	Login Landing	Login or Copy	/home/student/pbio8350/s_20 (Home) (Always!)
➔	Batch Job Submitting	Login or Interactive	/escratch4/s_20 (Scratch) (Suggested!) /home/student/pbio8350/s_20 (Home)
	Interactive Job Running	Interactive	/escratch4/s_20 (Scratch) /home/student/pbio8350/s_20 (Home)
	Data Archiving , Compressing and Transferring	Copy	/escratch4/s_20 (Scratch) /home/student/pbio8350/s_20 (Home)
	Job Data Temporarily Storing	Compute	/lscratch/s_20 (Local Scratch) /escratch4/s_146 (Scratch)

How does zcluster operate?

Next Page





How to work with zcluster?

Before we start:

- To get zcluster to be your best HPC buddy, go to
GACRC Wiki (<http://wiki.gacrc.uga.edu>)
GACRC Web (<http://gacrc.uga.edu>)
- To get the most effective and qualified support from us, go to
GACRC Support (https://wiki.gacrc.uga.edu/wiki/Getting_Help)
- To work happily and productively, follow the cluster's
Community Code of Conduct (**CCOC**)

How to work with it?

- Cluster's CCOC:

On cluster, you are not alone..... Each user is sharing finite resources, e.g., CPU cycles, RAM, disk storage, network bandwidth, with other researchers.

What you do may affect other researchers on the cluster.

6 rules of thumb to remember:

- NO jobs running on login node
- NO multi-threaded job running with only 1 core requested
- NO large memory job running on regular nodes
- NO long job running on interactive node
- NO small memory job running on large memory nodes
- Use the copy node for file transfer and compression



How to work with zcluster?

- Start with zcluster
- Connect & Login
- Transfer Files
- Software Installed
- Run Interactive Jobs
- Submit Batch Jobs
 - How to submit ***serial***, ***threaded***, and ***MPI*** batch jobs
 - How to check job status, cancel a job, etc.

How to work with zcluster – Start with zcluster

- You need a **Class User Account** : `s_20@zcluster.rcc.uga.edu`
- Procedure: https://wiki.gacrc.uga.edu/wiki/User_Accounts
- A UGA faculty member (**PI**) may register a computing lab:
<http://help.gacrc.uga.edu/labAcct.php>
- The PI of a computing lab may request user accounts for members of his/her computing lab: <http://help.gacrc.uga.edu/userAcct.php>
- User receives an email notification once the account is ready
- User can use `passwd` command to change initial temporary password

How to work with zcluster – Connect & Login

- Open a connection: Open a terminal and **ssh** to your account

```
ssh s_20@zcluster.rcc.uga.edu
```

or

```
ssh -X s_20@zcluster.rcc.uga.edu
```

⁽¹⁾ **-X** is for X windows application running on the cluster to be forwarded to your local machine

⁽²⁾ If using Windows, use **SSH client** to open connection, get from UGA download software page)

- Logging in: You will be prompted for your **zcluster password**

```
s_20@zcluster.rcc.uga.edu's password: █
```

⁽³⁾ On Linux/Mac, when you type in the password, the prompt blinks and does not move)

- Logging out: **exit** to leave the system

```
s_20@zcluster:~$ exit
```

How to work with zcluster – Transfer Files



- On Linux, Mac or cygwin on Windows : `scp [Source] [Target]`

E.g. 1: On local machine, do Local → zcluster

```
scp file1 s_20@copy.rcc.uga.edu:/escratch4/s_20/s_20_Jan_15/
```

```
scp *.dat s_20@copy.rcc.uga.edu:/escratch4/s_20/s_20_Jan_15/
```

E.g. 2: On local machine, do zcluster → Local

```
scp s_20@copy.rcc.uga.edu:/escratch4/s_20/s_20_Jan_15/file1 ./
```

```
scp s_20@copy.rcc.uga.edu:/escratch4/s_20/s_20_Jan_15/*.dat ./
```

- On Window: [FileZilla](#), [SSH Secure Client File Transfer](#), [WinSCP](#), etc.

How to work with zcluster – Softwares Installed

- Perl, Python, Java, awk, sed, C/C++ and Fortran compilers
- Matlab, Maple, R
- Many Bioinformatics applications: NCBI Blast+, Velvet, Trinity, TopHat, MrBayes, SoapDeNovo, SAMtools, **Mafft**, **RAxML**, **PASTA**, **MrBayes** and **MP-EST** etc.
- RCCBatchBlast (RCCBatchBlastPlus) to distribute NCBI Blast (NCBI Blast+) searches to multiple nodes.
- Many Bioinformatics Databases: NCBI Blast, Pfam, uniprot, etc.
- For a complete list of applications installed:
<https://wiki.gacrc.uga.edu/wiki/Software>

How to work with zcluster – Run Interactive Jobs

- From login node, **qlogin** command to open an **interactive** session:

Login Node: **zcluster.rcc.uga.edu** ^{qlogin}  Interactive Node: **compute-14-7/9**

```
s_20@zcluster:~$ qlogin
Your job 1391816 ("QLOGIN") has been submitted
waiting for interactive job to be scheduled ...
Your interactive job 1391816 has been successfully scheduled.
...
compute-14-7.local$
```

← Now I am on compute-14-7, which is an interactive node

- Current maximum runtime is **12** hours
- When you are done, **exit** to log out!
- Detailed information, like interactive parallel job? Go to:
https://wiki.gacrc.uga.edu/wiki/Running_Jobs_on_zcluster

How to work with zcluster – Submit Batch Jobs

- Components you need to submit a batch job:
 - **Software** already installed on zcluster (**Mafft**, **RAxML**, **PASTA** etc.)
 - **Job submission script** to run the software,
 - ✓ Specifying working directory
 - ✓ Exporting environment variables, e.g.,
OMP_NUM_THREADS (OpenMP threads number)
LD_LIBRARY_PATH (searching paths for shared libraries)
- Common commands you need:
 - **qsub** with specifying **queue name**, **threads** or **MPI rank number**
 - **qstat**, **qdel**
 - **qacct**, **qsj**, etc.

How to work with zcluster – Batch *Serial* Job

- **Step 1:** Create a job submission script *pt.sh* running PASTA:

```
#!/bin/bash                                → Linux default shell (bash)
cd /escratch4/s_20/s_20_Jan_15             → Specify and enter (cd) working directory (/escratch4/s_20/s_20_Jan_15)
unset MAFFT_BINARIES
export PATH=/usr/local/python/2.7.2/bin/:${PATH}
time python2.7 /usr/local/pasta/1.6.4/run_pasta.py [options] → Run pasta with 'time' command to measure amount of time it takes to run the application
```

- **Step 2:** Submit *pt.sh* to the queue:

```
$ qsub -q rcc-30d pt.sh
```

OR

```
$ qsub -q rcc-30d -l mem_total=20g pt.sh
```

Submit a job
to the queue rcc-30d
with **16GB** RAM/Node

job submission script

to the queue rcc-30d
with **48GB** RAM/Node

How to work with zcluster – Batch *Threaded* Job

- **Step 1:** Create a job submission script *mf.sh* running Mafft:

```
#!/bin/bash

cd / escratch4/s_20/s_20_Jan_15

export PATH=/usr/local/mafft/7.215-e/bin/:${PATH}

time mafft --thread 4 in.fasta > out → Run mafft with 4 threads
```

- **Step 2:** Submit *mf.sh* to the queue:

```
$ qsub -q rcc-30d -pe thread 4 mf.sh
```

4 cores requested

Threads Number = Number of Cores Requested

Note:

Please use the **rcc-mc-30d** queue,
If using threads **more than 8!**

How to work with zcluster – Batch *MPI* Job

- **Step 1:** Create a job submission script *raxml.sh* running RAxML:

```
#!/bin/bash

cd / escratch4/s_20/s_20_Jan_15

export MPIRUN=/usr/local/mpich2/1.4.1p1/gcc 4.5.3/bin/mpirun → Define and export environment variable (MPIRUN)

$MPIRUN -np $NSLOTS /usr/local/raxml/latest/raxmlHPC-MPI-SSE3 [options] → Run RAxML with 20 MPI processes (-np $NSLOTS )
```

- **Step 2:** Submit *raxml.sh* to the queue:

```
$ qsub -q rcc-30d -pe mpi 20 raxml.sh
```

20 cores requested,
\$NSLOTS will be assigned to 20 automatically, before
the job submission script is interpreted

How to work with zcluster – Check and Cancel Jobs

- To check the status of all queued and running jobs: **qstat**

<code>qstat</code>	→ shows your job in the pool
<code>qstat -u "*"</code>	→ shows all the jobs in the pool
<code>qstat -j 12345</code>	→ shows detailed information, e.g., maxvmem , about the job with JOBID 12345
<code>qstat -g t</code>	→ list all nodes used by your jobs

- To cancel a queued or running job: **qdel**

<code>qdel -u zhuofei</code>	→ deleted all your jobs
<code>qdel 12345</code>	→ deletes your job with JOBID 12345

- To list detailed information about a job: **qsj**, **qacct**

<code>qsj 12345</code>	→ shows information, e.g., maxvmem , about the RUNNING job with JOBID 12345
<code>qacct -j 12345</code>	→ shows information, e.g., maxvmem , about the ENDED job with JOBID 12345

Thank You!